

Weekly Report

胡万祺

一、 本周工作

【VisComposer】

理清几个主要模块的逻辑：

- Build

文本形式的代码需要通过 `viscomposer.Environment.build()` 进行编译。`build` 函数本质就是一个拥有一个闭包环境（和其中的私有变量、函数）、且将参数直接转接到 `eval` 函数。这样做可以为代码提供一些额外的便利，如可以使用一些常用函数 `min/max` 等，能简化用户编写代码时的代码量（否则可能需要用类似 `viscomposer.Math.min` 这样的形式）；另外还能控制代码运行所处的环境（如屏蔽掉可能造成不良影响的全局变量）。

- handler

`handler` 函数本质拥有一个闭包环境（和其中的私有变量、函数，预定义常用的交互过程），且根据参数将不同响应的回调函数返回的一个函数。

workflowHeader

这个模块主要是用来连接一个节点 `workflow` 和其子节点 `workflow` 的中间层对象，是属于其子节点的成员变量，主要负责定义数据分发的规则。

- SceneGraph

代表整个场景的一个模块，负责新建节点、加载模板节点等；其中有个 `run` 函数先删除所有图形，再重新添加 `scenegraph` 自己、根节点、根节点的儿子等等，添加之前先对所有节点进行拓扑排序。

另外还有几个单独的逻辑

- 保存模板：按照场景图结构保存，每层节点用 `json` 对象保存自己的关键几个属性，并调用自己所有子节点的保存模板的函数。`Node` 为模板保存的最高级别的对象，往下是 `workflow, workflowheader, module/link, port`
- 载入模板：按层解析 `json` 对象，逐层新建相应的对象并恢复所有属性配置。
- 更新：按照场景图结构进行逐层的更新，更新顺序为 `Scenegraph, node, workflow, workflowheader, module, link, port`

其他的都是细节，比如每个具体的功能模块里面一些逻辑的实现。

【农业大数据】

陈俊：

- 需求文档补充

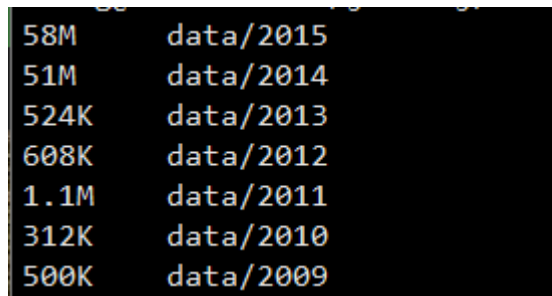
在之前的需求说明书的基础之上，补充了一些内容，主要是一些图的补充，包括用例图、状态图、数据流图，目前来说比较粗糙，具体细节部分还有待讨论，尤其一些输入输出的部分尚不明确。

- 爬去 3w3n 网站数据

爬取了 3w3n 这个网站的数据，主要使用 `nodejs` 与 `jquery` 的库，做的时候遇到了编码问

题，原以为 nodejs 都是默认 utf-8 的，没想到还是显式指定一下比较好。

这个网站数据分布不是很均匀。每年的分布如下（其中 15 年的数据只到 7 月份）



58M	data/2015
51M	data/2014
524K	data/2013
608K	data/2012
1.1M	data/2011
312K	data/2010
500K	data/2009

当然可以看到这两年的数据相比之前稳定一些，经过一些观察，这个网站一般来说每天会更新约 1000 条记录，但是大约间隔一个月左右会有一次高峰期，一天可以更新数万条记录，原因不明。

这个网站爬起来相对自由一些，可以指定时间段爬（按天），当然每天一爬也是可以做到的。

王艺：

- PGSQL 集群搭建

因为 PG 本身不支持集群，并且集群的资料也比较少，所以在尝试各种第三方扩展做集群搭建，目前没有满意的方案。

详细请看博客：[Postgres-xl 集群搭建](#)

二、 下周任务：

- 1、VisComposer 整个流程跑通。
- 2、pgsql 集群系统搭建完成